



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/015,378	12/12/2001	James Sheung Lau	CA920000074US1 (7161-208U)	2828
46320 7590 06/27/2007 CAREY, RODRIGUEZ, GREENBERG & PAUL, LLP STEVEN M. GREENBERG 950 PENINSULA CORPORATE CIRCLE SUITE 3020 BOCA RATON, FL 33487			EXAMINER SIMITOSKI, MICHAEL J	
			ART UNIT 2134	PAPER NUMBER
			MAIL DATE 06/27/2007	DELIVERY MODE PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.



UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

MAILED

JUN 27 2007

Technology Center 2100

**BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES**

Application Number: 10/015,378
Filing Date: December 12, 2001
Appellant(s): LAU, JAMES SHEUNG

Scott D. Paul
Registration No. 42,984
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 2/27/2007 appealing from the Office action mailed 9/26/2006.

Art Unit: 2134

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

6,070,171	SNYDER et al.	5-2000
6,748,362	MEYER et al.	1-2004
5,946,677	BULLEN	8-1999
6,784,925	TOMAT et al.	8-2004

Art Unit: 2134

COLVIN, David S. U.S. Patent Application Publication 2004/0225900 A1, 11-2004.

PEINADO, Marcus et al. U.S. Patent Application Publication 2003/0078853 A1, 4-2003.

STORY JR, Guy A. et al. U.S. Patent Application Publication 2002/0046181, 4-2002.

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claim Rejections - 35 USC § 103

1. Claims 1-13, 15, 19-20, 29 & 31-35 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent 6,070,171 to Snyder et al. (**Snyder**) in view of U.S. Patent 6,748,362 to Meyer et al. (**Meyer**).

Regarding claims 1, 3, 6, 10-11, 15, 19-20, 29 & 32-35, Snyder discloses embedding in computer readable content (software, col. 2, lines 25-28 & col. 12, lines 21-25), instruction codes (tracker client program, col. 5, lines 44-50) operable to direct a processor circuit to automatically establish a connection to a server (col. 12, lines 28-30), when said content (software) is in use by said processor circuit (col. 12, lines 28-30), to transmit registration information (token) to said server (col. 12, lines 28-30) and operable to control further use of said content (software) by said processor circuit (Fig. 3a, #48) in response to a key (new token) and permission received from said server (col. 12, lines 36-37 & Fig. 3a, #48). Snyder lacks the content being non-functional descriptive content. However, Meyer teaches that non-functional descriptive content (media file, audio, etc., col. 4, lines 32-40 & col. 5, line 59 – col. 1, line 3) can have embedded therein executable code or application-specific data (col. 6, lines 43-55) for the purposes of execution of the code simultaneously with the use of the non-functional descriptive content to control the exact behavior of the execution environment relative to the content while playback is occurring

Art Unit: 2134

(col. 10, lines 42-53). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to modify Snyder to embed the Tracker client program and other required data into non-functional descriptive content. One of ordinary skill in the art would have been motivated to perform such a modification to control the exact behavior of the execution environment of the content and track usage of the content, as taught by Meyer (col. 4, lines 32-40, col. 5, line 59 – col. 1, line 3, col. 6, lines 43-55 & col. 10, lines 42-53).

Regarding claims 2, 7 & 9, Snyder, as modified above, discloses a portable memory medium (CD-ROM, col. 11, lines 34-42 & col. 12, lines 12-15).

Regarding claims 4-5 & 31, Snyder, as modified above, discloses a self-executing applet (tracker client, col. 12, lines 12-15).

Regarding claim 8, Snyder, as modified above, discloses transmitting said computer readable content and said embedded instruction codes on a communications network (email, col. 12, lines 56-58).

Regarding claims 12-13, Snyder, as modified above, discloses producing a measure of use of said content (current token, col. 7, lines 20-21).

2. Claims 1-11, 16-20, 39 & 31-35 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent Application Publication 2004/0225900 to **Colvin** in view of **Meyer**. Colvin teaches embedding, in computer readable content, instruction codes (§29) operable to direct a processor circuit to automatically establish a connection to a server (§29 & §32), when said content is in use by said processor circuit, to transmit registration information to said server (§32), and operable to control further use of said content by said processor circuit in

Art Unit: 2134

response to a key/authorization code (§29 & §32-33), wherein the instruction codes (software) include self-executing (installation) application code. Colvin lacks the content being non-functional descriptive content. However, Meyer teaches that non-functional descriptive content (col. 4, lines 32-40 & col. 5, line 59 – col. 1, line 3) can have embedded therein executable code or application-specific data (col. 6, lines 43-55) for the purposes of execution of the code simultaneously with the use of the non-functional descriptive content to control the exact behavior of the execution environment relative to the content while playback is occurring (col. 10, lines 42-53). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to modify Colvin to embed the Tracker client program and other required data into non-functional descriptive content. One of ordinary skill in the art would have been motivated to perform such a modification to control the exact behavior of the execution environment of the content and track usage of the content, as taught by Meyer (col. 4, lines 32-40, col. 5, line 59 – col. 1, line 3, col. 6, lines 43-55 & col. 10, lines 42-53).

3. Claims 16-17 are rejected under 35 U.S.C. 103(a) as being unpatentable over **Snyder** and **Meyer**, as applied to claim 12 above, in further view of U.S. Patent Application Publication 2003/0078853 to Peinado et al. (**Peinado**).

Regarding claim 16, Snyder lacks establishing a connection to said server when said measure of use exceeds a threshold value. However, Peinado teaches by including a license with a description of constraints to be met before content is rendered (§18), a user is restricted to a limited number of uses of content (§128 & §136). Peinado further discloses that the user can obtain additional access to the content by contacting the server to obtain a valid license (§137 & §140). Therefore, it would have been obvious to one having ordinary skill in the art at the time

Art Unit: 2134

the invention was made to modify Snyder to contact the server when a measure exceeds a threshold. One of ordinary skill in the art would have been motivated to perform such a modification to limit the number of uses of the content, as taught by Peinado (§18, 128, 133-137 & 140).

Regarding claim 17, Snyder discloses an Internet protocol connection (col. 4, lines 40-43).

4. Claim 18 is rejected under 35 U.S.C. 103(a) as being unpatentable over **Snyder, Meyer and Peinado**, as applied to claim 17 above, in further view of U.S. Patent Application Publication 2002/0046181 to Story, Jr. et al. (**Story**). Snyder, as modified above, discloses a browse session with a uniform resource locator/web site pointing to a user registration page (§140-§141), but lacks explicitly the page permitting a user to enter registration information. However, Story teaches that it is known to use a web page to authenticate a user before supplying the user with a license (§39). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to modify Snyder to use the web page supplied by the server to permit the user to enter registration information. One of ordinary skill in the art would have been motivated to perform such a modification to authenticate the user before authorizing the creation of a license, as taught by Story (§39).

5. Claim 21 is rejected under 35 U.S.C. 103(a) as being unpatentable over **Snyder and Meyer**, as applied to claim 20 above, in further view of U.S. Patent 5,946,677 to **Bullen**. Snyder, as modified above, lacks deleting files produced by functional descriptive content in said

Art Unit: 2134

computer readable content. However, Bullen teaches that it is well known for application programs to generate temporary files that are generally unneeded and which must later be deleted to free up additional storage capacity (col. 2, lines 12-15). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to modify Snyder to delete files produced by functional descriptive content/tracker client and software in said computer readable content/software. One of ordinary skill in the art would have been motivated to perform such a modification to free up additional storage capacity, as taught by Bullen (col. 2, lines 12-15).

6. Claim 22 is rejected under 35 U.S.C. 103(a) as being unpatentable over **Snyder, Meyer and Bullen**, as applied to claim 21 above, in further view of U.S. Patent 6,784,925 to Tomat et al. (**Tomat**). Snyder, as modified above, lacks warning a user of said processor circuit that files are about to be deleted. However, Tomat teaches that when temporary disk space runs low, it is beneficial to warn a user that files need to be deleted (col. 23, lines 11-16). Therefore, it would have been obvious to one having ordinary skill in the art at the time the invention was made to modify Snyder to warn the user when files are about to be deleted. One of ordinary skill in the art would have been motivated to perform such a modification to allow the user to choose to modify the temporary storage space, as taught by Tomat (col. 23, lines 11-16).

(10) Response to Argument

Appellant's brief (§VII)

Appellant's brief (p. 6) argues the following:

“Upon reviewing this passage, Appellant is unclear as to where Meyer precisely teaches using the embedded executable code “to control the exact behavior of the execution environment of the content and track usage of the content.” Appellant has been unable to find any mention of tracking usage of the content within Meyer. Moreover, referring to the Examiner's cited passage reproduced above, Meyer teaches that the Application Programming Interfaces (APIs) “control the exact behavior of the execution environment.” Meyer also teaches that an “execution environment is then invoked to begin execution of the execution code.” However, Appellant is unclear as to where Meyer specifically teaches that the executable code stored within the content controls the APIs. Appellant, therefore, respectfully submits that Meyer cannot be relied upon to teach using the executable code within content to control the execution environment of the content, as asserted by the Examiner.”

Attention is drawn to the following quotations, with emphasis added, from Meyer in describing the invention disclosed therein:

(col. 4, lines 7-18) “Our techniques allow for media files to contain entirely new classes of content, such as: entire computer programs, multimedia annotations, or lengthy supplemental communications. As described in said copending application, computer programs embedded in media files allow for expanded integrated transactional media of all kinds, including merchandising, interactive content, interactive and traditional advertising, polls, e-commerce solicitations such as CD or concert ticket purchases, and fully reactive content such as games and interactive music videos which react to the user's mouse motions and are synced to the beat of the music.”

(col. 4, lines 23-28) “This invention even creates new business models. For example, instead of a record company trying to stop the copying of its songs, it might instead encourage the free and open distribution of the music, so that the embedded advertising and e-commerce messages are spread to the largest possible audience of potential customers.”

(col. 5, lines 46-49) “This is done, moreover, in such a manner as to enable such supplemental data to be extracted by digital media playback apparatus, while preserving backwards compatibility with existing media players.”

(col. 10, lines 9-15) “This header is placed at the beginning of the executable content. The entire package, of the header plus the content, is then transformed into a bit-stream by examining each octet in turn, and placing its bits into the bit-stream from Most Significant-Bit to Least-Significant-Bit ordering. Now that the bit-stream has been generated, it is necessary to embed it into the MPEG audio file.”

(col. 10, lines 34-53) “... the decoding process is a simple reversal of the encoding process above-detailed. The first octet is extracted from the audio data, and based on that,

all succeeding bits are extracted as well. If there it is a checksum encoded in the header, the octets of the executable code are summed, the result taken modulo 256, and compared with the encoded checksum. If they are equal, then execution proceeds.

Based on the MIME type of the executable code, an appropriate execution environment is instantiated. In the case of the application/x-shockwave-flash type discussed previously, a reference execution environment is described by Macromedia in the Flash Standards web page.

The execution environment is then invoked to begin execution of the executable code simultaneously with the playback of the audio file. Additional Application Programming Interfaces (APIs) may be defined with reference to the execution environment to control the exact behavior of the execution environment relative to the audio file while playback is occurring.”

Meyer's invention encodes executable content such as a computer program in non-functional descriptive content (i.e. a media file) such that when the media content is used/explored/played, etc., the executable content embedded therein will execute accordingly. Snyder's invention tracks software by embedding additional software (tracker client program) within the parent software (col. 5 of Snyder gives the example of a stock analysis program) the tracker software performing, among other things, a connection to a server; the end result is the tracking of the stock analysis software (see Snyder col. 5, lines 20-59). In combination, one of ordinary skill in the art finds motivation to modify Snyder's invention such that the tracker client program is embedded not within a parent program, as disclosed by Snyder, but within non-functional descriptive content such as a media file, by using the Meyer invention. This usefulness of this modification is clear from Meyer disclosing that, when executing the embedded program within the non-functional descriptive content, an execution environment is instantiated and used, via APIs, to control the exact behavior of the execution environment of the non-functional descriptive content. Further motivation is seen in col. 2, lines 17-21 of Meyer, which teaches that the embedded data supplements the non-functional descriptive content with e-commerce content presentations at the digital playback device (note again that Snyder's

Art Unit: 2134

invention supplements functional descriptive content with an e-commerce application in the form of tracking software). Further explanation is given in the following paragraph.

Turning to the last quotation above from Meyer, by executing the executable code (which has been embedded in the non-functional descriptive media), an appropriate execution environment is instantiated; Meyer gives the example of application/x-shockwave-flash. Meyer is disclosing that by embedding an x-shockwave/flash application in the media content, all the API's (function calls, variables, abilities) associated with a flash file are now available to the executing embedded code, (i.e. the application/x-shockwave-flash type file embedded within the audio file). Meyer's statement ("Additional Application Programming Interfaces (APIs) may be defined with reference to the execution environment to control the exact behavior of the execution environment relative to the audio file while playback is occurring") teaches what one having ordinary skill in the art would recognize as a strong motivation to use Meyer's embedding process. When the executable code within the audio file is executed, the necessary environment is instantiated and it is that environment that can be used to control the exact behavior of the execution environment relative to the audio file. This environment, in the proposed modification, performs the usage tracking functions of Snyder. The audio file is now under control of the execution environment, which receives its commands from the embedded code. In the shockwave/flash example, one having ordinary skill in the art knows that a flash/shockwave file is a file type that, when executed, creates an instance (instantiates) the flash/shockwave engine, which interprets code in the flash/shockwave file such that the user's device performs functions according to the flash/shockwave file. This is similar to another familiar product – Java – where executing a Java program instantiates the Java Virtual Machine,

Art Unit: 2134

which interprets the code in the Java program and causes the user's computer to perform the functions specified in the Java program. It is again noted that the main purpose of the Snyder invention is to track and control the usage of software packages after those packages have been distributed to known or unknown users (Snyder, col. 5, lines 20-26). Appellant suggests that Meyer cannot be relied upon to teach using the executable code within the content to control the execution environment of the content because "Appellant is unclear as to where Meyer specifically teaches that the executable code stored within the content controls the API's".

However, as described above, the API's associated with the execution environment are used by the embedded code under consideration; in the flash and Java examples above, the flash and Java APIs (through which flash and Java programs communicate to the flash and Java engines) are used for responding to flash and Java commands, respectively. Meyer's disclosure teaches that these APIs can be used to control the execution of the audio file. Therefore, motivation exists to embed the usage tracking software of Snyder in non-functional descriptive content in a manner as taught by Meyer to allow the usage tracking software of Snyder to track the usage of non-functional descriptive content by controlling the behavior of the non-functional descriptive content. This is similar to the usage tracking software controlling the behavior of Snyder's program-to-be-tracked.

Appellant's brief (p. 7, ¶2) argues that Meyer's teaching is a generalization that is silent as to how the execution environment of the content is controlled and for what purpose the execution environment is controlled. However, in the proposed combination, Meyer's non-functional descriptive content (when used/played, etc.) would perform the usage-tracking functions of Snyder. As such, the proposed combination only requires one having ordinary skill

Art Unit: 2134

in the art to take the usage tracking code of Snyder's invention and use Meyer's embedding process to embed this usage tracking code in a media file (non-functional descriptive content) of choice. The Examiner submits that combining the two inventions in such a manner flows naturally from the respective descriptions, needing no additional guidance.

Appellant's brief (p. 7, ¶3) argues that neither of Snyder nor Meyer recognized the problems described by the instant application. However, the fact that appellant has recognized another advantage which would flow naturally from following the suggestion of the prior art cannot be the basis for patentability when the differences would otherwise be obvious. See *Ex parte Obiaya*, 227 USPQ 58, 60 (Bd. Pat. App. & Inter. 1985).

Appellant's brief (pp. 8-11) indicates that claims 2-11, 16-20, 29 and 31-35 rely upon the argument given for the rejection of claim 1. As the arguments given with respect to claim 1 have been addressed above, the Examiner relies upon the above response in consideration of claims 2-11, 16-20, 29 and 31-35.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

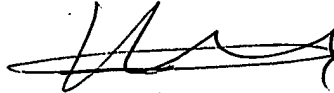
Art Unit: 2134

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

Michael J. Simitoski

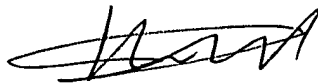
/Michael J. Simitoski/



KAMBIZ ZAND
SUPERVISORY PATENT EXAMINER

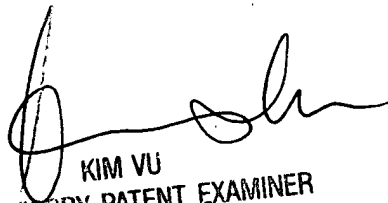
Conferees:

Kambiz Zand



KAMBIZ ZAND
SUPERVISORY PATENT EXAMINER

Kim Vu



KIM VU
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100